

National Oceanic and Atmospheric Administration (NOAA)

*Reimbursable Agreement Summary Report
(RA502D) Specifications*

June 2, 2006

Table of Contents

1. REIMBURSABLE AGREEMENT SUMMARY REPORT (RA502D)	1
2. SELECTION PARAMETERS	2
2.1 BUTTONS	14
3. REPORT FORMAT	15
3.1 REPORT FIELD ATTRIBUTES	15
3.2 SORT ORDER, BREAKS, AND TOTALS	19
3.3 RA502 REPORT FORMAT	21
4. SELECTION CRITERIA AND REPORT EXECUTION	24
4.1 SINGLE CURSOR METHOD	25
4.2 TEMPORARY TABLE METHOD (ALTERNATIVE METHOD)	30
4.3 GETTING ORG DESCRIPTION, FUND TITLE	31
4.4 TITLE PAGE	31
5. DATA WAREHOUSE STRUCTURE SUPPORT	33
6. VIEWS	34
7. NAVIGATOR MENU PARAMETERS	37
8. NAVIGATOR REPORT HEADER AND COVER PAGE DEFINITIONS	38

1. Reimbursable Agreement Summary Report (RA502D)

The Reimbursable Module records and stores a large volume of financial data for each reimbursable agreement. The Reimbursable Agreement Summary Report (RA502D) has been designed to summarize the agreement information for the use of program managers and other personnel responsible for the overall financial management of the agreement. In addition to contractual data, such as period of performance and billing type, the report displays the total authorized order, collected advance (for advance projects), billed costs, costs collected, unreleased costs, uncollected advance (for advance projects) and the remaining order balance for a single reimbursable agreement. Undelivered order amounts and unmatched costs are not included on this report because they are not allocated to a reimbursable agreement until they are billed by the WIP process.

The following subsections describe the basic requirements for the RA502D report:

- Section 2 describes the selection parameters by which the user can request the report.
- Section 3 describes the data elements to appear on the report, the format of the report, and the sort order, breaks, and levels at which totals will be reflected.
- Section 4 describes the criteria to be used to select data and the logic used to execute the report.
- Section 5 identifies tables from CFS that must be snapshot into the data warehouse in order to generate this report.
- Section 6 describes the views that form the basis of the data gathering queries.
- Section 7 describes the use of the Navigator parameter table.
- Section 8 describes the standard conventions used for populating the Report Header and Cover Page for reports generated as data warehouse Navigator reports.

2. Selection Parameters

A parameter screen will be used to specify selection parameters and to generate the report. The parameter screen does not use a base table, it only uses non-database fields.

RA502D Reimbursable Agreement Summary Report [X]

Bureau Code: ## [V] bureau name

Fund Code: ALL [V] "All Funds" or fund title

Acceptance Org: ALL [V] - ALL [V] - ALL [V] - ALL [V] - ALL [V] - ALL [V] - ALL [V]

Agreement Status: ALL [V]

Agreement Number: ALL [V]

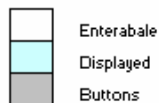
TwA Number: ALL [V]

Customer No: ALL [V] "All Customers" or Customer Name

As of GI End Date: dd-mon-yyyy [V] FY year

Order No Sequence: ASCENDING [V]

Exit Default Run Report



The fields in **red** indicate the default values or the default format for the field. The fields in white are enterable. The fields in blue are navigable but not enterable.

The following table specifies the attributes of each field. Upon startup all fields will be set to their default values.

Field Usage in RA502

Screen Name	Attribute	Specification
Bureau Code	Field Name	P_BUREAU_CODE
	Attributes	Enterable, Required, Displayed
	Type	Number(2) format "00"
	Default Value	Select EC.d_bureau_code,BU.bureau_Name from ndw_employee_v EC, Bureau BU where EC.D_bureau_code = BU.bureau_code and EC.db_user_name = user; Select into P_BUREAU_CODE and bureau name fields.

Field Usage in RA502

Screen Name	Attribute	Specification
	Validation	<pre>Select BU.bureau_Name from Bureau BU where BU.bureau_code = :BLOCK1.P_BUREAU_CODE And BI.active_status = 'Y'</pre> <p>Select into P_BUREAU and bureau name fields.</p> <p>Error Message: "Invalid Bureau Code."</p>
	LOV	<pre>Select BU.Bureau_Code,BU.bureau_Name from Bureau BU where BU.active_status = 'Y' order by 1;</pre>
Bureau Name	Field Name	Bureau_Name – Non-Database, Unlabeled
	Attributes	Not-Enterable, Navigable, Displayed
	Type	Varchar2(40) Default
	Default Value	Defaulted by startup query or by the validation routine for Bureau_Code (P_BUREAU_CODE)
Fund Code	Field Name	P_FUND_CODE
	Attributes	Enterable, Required, Displayed
	Type	Varchar2(4)
	Default Value	ALL
	LOV	<pre>Select 'ALL' fund_code, 'All Funds' fund_name , -1 orderkey From dual UNION Select lpad(FU.fund_code,4,0) , FU.fund_name , FU.fund_code from Fund FU where FU.fund_Type in (select parameter_value from cams_menus cmc, cams_menu_parameters cmp where upper(cmc.menu_code) = 'RA502D' and cmc.menu_id = cmp.menu_id and cmp.parameter_name = 'REIMB_FUND_TYPES') and FU.bureau_code = :BLOCK1.PN And FU.rowid = (select FU2.rowid from FUND FU2 where FU.bureau_code = FU2.bureau_code and FU.fund_code = FU2.fund_code and FU.begin_date <= Sysdate); order by 3;</pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>

Field Usage in RA502

Screen Name	Attribute	Specification
	Validation	Based on the LOV – the word ALL or a valid fund code. Required, Displayed. Error Message: “Invalid Fund Code.”
Fund Title	Field Name	Fund_Title – Non-Database, Unlabeled
	Attributes	Not-Enterable, Navigable, Displayed
	Type	Varchar2(40)
	Value	Defaulted value “All Funds”. See LOV on Fund Code (P_Fund_Code).
Acceptance Org	Overview	Seven fields whose values are dependent on the prior field values
Acceptance Org1	Field Name	P_ACCEPT_ORG1
	Attributes	Enterable, Required, Displayed
	Type	Varchar2(3)
	Default Value	ALL
	LOV	<pre> Select 'ALL' org1_code, 'All Organizations' Org1_descr , ' ' orderkey From dual UNION Select lpad(ORG1.org1_code,2,0) , ORG1.org1_descr , ORG1.org1_code from Org1 where ORG1.bureau_code = :BLOCK1.P_BUREAU_code order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>
	Validation	Based on the LOV – the word ALL or a valid org1 code. If the value “ALL” is entered default the value of P_ACCEPT_ORG2, P_ACCEPT_ORG3, P_ACCEPT_ORG4, P_ACCEPT_ORG5, P_ACCEPT_ORG6 and P_ACCEPT_ORG7 to “ALL”. Error Message: “Invalid Org 1 Code.”
Acceptance Org2	Field Name	P_ACCEPT_ORG2
	Attributes	Required, Displayed. Initially Not Enterable. Set to enterable if P_ACCEPT_ORG1 <> ‘ALL’.
	Type	Varchar2(3)
	Default Value	ALL

Field Usage in RA502

Screen Name	Attribute	Specification
	LOV	<pre> Select 'ALL' org2_code, 'ALL Org2 Codes' Org2_descr , ' ' orderkey From dual UNION Select lpad(ORG2.org2_code,2,0) , ORG2.org2_descr , ORG2.org2_code from Org2 where ORG2.bureau_code = :BLOCK1.P_BUREAU_code and :BLOCK1.P_ACCEPT_ORG1 <> 'ALL' and :BLOCK1.P_ACCEPT_ORG1 = ORG2.org1_code order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>
	Validation	<p>Based on the LOV – the word ALL or a valid org2 code.</p> <p>If the value “ALL” is entered default the value of P_ACCEPT_ORG3, P_ACCEPT_ORG4, P_ACCEPT_ORG5, P_ACCEPT_ORG6 and P_ACCEPT_ORG7 to “ALL”.</p> <p>Error Message: “Invalid Org 3 Code.”</p>
Acceptance Org3	Field Name	P_ACCEPT_ORG3
	Attributes	Required, Displayed. Initially Not Enterable. Set to enterable if P_ACCEPT_ORG2 <> 'ALL'.
	Type	Varchar2(4)
	Default Value	ALL
	LOV	<pre> Select 'ALL' Org3_code, 'ALL Org3 Codes' Org3_descr , ' ' orderkey From dual UNION Select lpad(ORG3.org3_code,4,0) , ORG3.Org3_descr , ORG3.Org3_code from Org3 where ORG3.bureau_code = :BLOCK1.P_BUREAU_code and :BLOCK1.P_ACCEPT_ORG2 <> 'ALL' and :BLOCK1.P_ACCEPT_ORG1 = ORG3.org1_code and :BLOCK1.P_ACCEPT_ORG2 = ORG3.org2_code order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>
	Validation	<p>Based on the LOV – the word ALL or a valid org3 code.</p> <p>If the value “ALL” is entered default the value of P_ACCEPT_ORG4, P_ACCEPT_ORG5, P_ACCEPT_ORG6 and P_ACCEPT_ORG7 to “ALL”.</p> <p>Error Message: “Invalid Org 3 Code.”</p>
Acceptance Org4	Field Name	P_ACCEPT_ORG4
	Attributes	Enterable, Required, Displayed

Field Usage in RA502

Screen Name	Attribute	Specification
	Type	Varchar2(3)
	Default Value	ALL
	LOV	<pre> Select 'ALL' Org4_code, 'ALL Org4 Codes' Org4_descr , ' ' orderkey From dual UNION Select lpad(ORG4.org4_code,2,0) , ORG4.Org4_descr , ORG4.Org4_code from Org4 where ORG4.bureau_code = :BLOCK1.P_BUREAU_code and :BLOCK1.P_ACCEPT_ORG3 <> 'ALL' and :BLOCK1.P_ACCEPT_ORG1 = ORG4.org1_code and :BLOCK1.P_ACCEPT_ORG2 = ORG4.org2_code and :BLOCK1.P_ACCEPT_ORG3 = ORG4.org3_code order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>
	Validation	<p>Based on the LOV – the word ALL or a valid org4 code.</p> <p>If the value “ALL” is entered default the value of P_ACCEPT_ORG5, P_ACCEPT_ORG6 and P_ACCEPT_ORG7 to “ALL”.</p>
Acceptance Org5	Field Name	P_ACCEPT_ORG5
	Attributes	Required, Displayed. Initially Not Enterable. Set to enterable if P_ACCEPT_ORG4 <> 'ALL'.
	Type	Varchar2(3)
	Default Value	ALL
	LOV	<pre> Select 'ALL' Org5_code, 'ALL Org5 Codes' Org5_descr , ' ' orderkey From dual UNION Select lpad(ORG5.org5_code,2,0) , ORG5.Org5_descr , ORG5.Org5_code from Org5 where ORG5.bureau_code = :BLOCK1.P_BUREAU_code and :BLOCK1.P_ACCEPT_ORG4 <> 'ALL' and :BLOCK1.P_ACCEPT_ORG1 = ORG5.org1_code and :BLOCK1.P_ACCEPT_ORG2 = ORG5.org2_code and :BLOCK1.P_ACCEPT_ORG3 = ORG5.org3_code and to_number(:BLOCK1.P_ACCEPT_ORG4) = ORG5.org4_code order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV.</p> <p>If the value “ALL” is entered default the value of P_ACCEPT_ORG6 and P_ACCEPT_ORG7 to “ALL”.</p>

Field Usage in RA502

Screen Name	Attribute	Specification
	Validation	Based on the LOV – the word ALL or a valid org5 code. If the value “ALL” is entered default the value of P_ACCEPT_ORG6 and P_ACCEPT_ORG7 to “ALL”. Error Message: “Invalid Org 5 Code.”
Acceptance Org6	Field Name	P_ACCEPT_ORG6
	Attributes	Required, Displayed. Initially Not Enterable. Set to enterable if P_ACCEPT_ORG5 <> ‘ALL’.
	Type	Varchar2(3)
	Default Value	ALL
	LOV	<pre> Select 'ALL' Org6_code, 'ALL Org6 Codes' Org6_descr , ' ' orderkey From dual UNION Select lpad(ORG6.org6_code,2,0) , ORG6.Org6_descr , ORG6.Org6_code from Org6 where ORG6.bureau_code = :BLOCK1.P_BUREAU_code and :BLOCK1.P_ACCEPT_ORG4 <> 'ALL' and :BLOCK1.P_ACCEPT_ORG1 = ORG6.org1_code and :BLOCK1.P_ACCEPT_ORG2 = ORG6.org2_code and :BLOCK1.P_ACCEPT_ORG3 = ORG6.org3_code and to_number(:BLOCK1.P_ACCEPT_ORG4) = ORG6.org4_code and to_number(:BLOCK1.P_ACCEPT_ORG5) = ORG6.org5_code order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>
	Validation	Based on the LOV – the word ALL or a valid org6 code. If the value “ALL” is entered default the value of P_ACCEPT_ORG7 to “ALL”. Error Message: “Invalid Org 6 Code.”
Acceptance Org7	Field Name	P_ACCEPT_ORG7
	Attributes	Required, Displayed. Initially Not Enterable. Set to enterable if P_ACCEPT_ORG6 <> ‘ALL’.
	Type	Varchar2(3)
	Default Value	ALL

Field Usage in RA502

Screen Name	Attribute	Specification
	LOV	<pre> Select 'ALL' Org7_code, 'ALL Org7 Codes' Org7_descr , ' ' orderkey From dual UNION Select lpad(ORG7.org7_code,2,0) , ORG7.Org7_descr , ORG7.Org7_code from Org7 where ORG7.bureau_code = :BLOCK1.P_BUREAU_code and :BLOCK1.P_ACCEPT_ORG4 <> 'ALL' and :BLOCK1.P_ACCEPT_ORG1 = ORG7.org1_code and :BLOCK1.P_ACCEPT_ORG2 = ORG7.org2_code and :BLOCK1.P_ACCEPT_ORG3 = ORG7.org3_code and to_number(:BLOCK1.P_ACCEPT_ORG4) = ORG7.org4_code and to_number(:BLOCK1.P_ACCEPT_ORG5) = ORG7.org5_code and to_number(:BLOCK1.P_ACCEPT_ORG6) = ORG7.org6_code order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>
	Validation	<p>Based on the LOV – the word ALL or a valid org7 code.</p> <p>Error Message: “Invalid Org 7 Code.”</p>
Agreement Status	Field Name	P_STATUS
	Attributes	Enterable, Required, Displayed
	Type	Varchar2(6)
	Default Value	ALL
	LOV	<pre> Select 'ALL' code_value, 'All Open or Active Agreements' code_descr , ' ' orderkey From dual UNION Select Code_Value, Code_descr , code_value from Accounting_Code where code_type = 'RASTAT' and active_status = 'Y' order by 3; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV</p>
	Validation	<p>Based on the LOV – the word ALL or a valid code_value.</p> <p>Error Message: “Invalid Reimbursable Agreement Status code.”</p>
Agreement Number	Field Name	P_AGREEMENT_NO
	Attributes	Enterable, Required, Displayed
	Type	Varchar2(30)
	Default Value	ALL

Field Usage in RA502

Screen Name	Attribute	Specification
	LOV	<pre> Select 'ALL' Agreement_No , null Customer_No , 'All Customers' Customer_Name , ' ' orderkey From dual UNION Select RA.agreement_no , RA.customer_no , CC.customer_name , RA.agreement_no from CUSTOMER CC, RA_CONTROL RA where RA.bureau_code = :BLOCK1.P_BUREAU_code and CC.customer_no=RA.customer_no and (:BLOCK1.P_FUND_CODE='ALL' or :BLOCK1.P_FUND_CODE = RA.fund_code) and (:BLOCK1.P_ACCEPT_ORG1='ALL' or :BLOCK1.P_Accept_Org1 = RA.accept_org1_code) and (:BLOCK1.P_ACCEPT_ORG2='ALL' or :BLOCK1.P_Accept_Org2 = RA.accept_org2_code) and (:BLOCK1.P_ACCEPT_ORG3='ALL' or :BLOCK1.P_Accept_Org3 = RA.accept_org3_code) and (:BLOCK1.P_ACCEPT_ORG4='ALL' or :BLOCK1.P_Accept_Org4 = RA.accept_org4_code) and (:BLOCK1.P_ACCEPT_ORG5='ALL' or :BLOCK1.P_Accept_Org5 = RA.accept_org5_code) and (:BLOCK1.P_ACCEPT_ORG6='ALL' or :BLOCK1.P_Accept_Org6 = RA.accept_org6_code) and (:BLOCK1.P_ACCEPT_ORG7='ALL' or :BLOCK1.P_Accept_Org7 = RA.accept_org7_code) and RA.approved_flag='Y' and RA.formal_flag='Y' and RA.trans_id = (Select max(RA2.trans_id) from RA_CONTROL RA2 where RA.bureau_code=RA2.bureau_code and RA.document_o = RA2.document_no and RA.formal_flag='Y' and RA2.approved_flag = 'Y') order by 4; </pre> <p>The orderkey is not displayed in the LOV. It is only used for ordering the LOV.</p> <p>The Customer_No and Customer_Name are copies to the screen fields (P_Customer_No and Customer_Name), if the Agreement No (P_AGREEMENT_NO) is not ALL.</p>
	Validation	<p>Based on the LOV – the word ALL or a valid Agreement Number.</p> <p>Error Message: “Invalid Agreement Number.”</p>

Field Usage in RA502

Screen Name	Attribute	Specification
TWA Number	Field Name	P_TWA_NO
	Attributes	Enterable, Required, Displayed
	Type	Varchar2(30)
	Default Value	ALL
	LOV	<pre> Select 'ALL' TEMP_WORK_AUTH_NO Select 'ALL' Temp_Word_Auth_No , null Customer_No , 'All Customers' Customer_Name , ' ' orderkey From dual UNION Select RA.agreement_no , RA.customer_no , CC.customer_name , RA.agreement_no from CUSTOMER CC, RA_CONTROL RA where RA.bureau_code = :BLOCK1.P_BUREAU_code and CC.customer_no=RA.customer_no and (:BLOCK1.P_FUND_CODE='ALL' or :BLOCK1.P_FUND_CODE = RA.fund_code) and (:BLOCK1.P_ACCEPT_ORG1='ALL' or :BLOCK1.P_Accept_Org1 = RA.accept_org1_code) and (:BLOCK1.P_ACCEPT_ORG2='ALL' or :BLOCK1.P_Accept_Org2 = RA.accept_org2_code) and (:BLOCK1.P_ACCEPT_ORG3='ALL' or :BLOCK1.P_Accept_Org3 = RA.accept_org3_code) and (:BLOCK1.P_ACCEPT_ORG4='ALL' or :BLOCK1.P_Accept_Org4 = RA.accept_org4_code) and (:BLOCK1.P_ACCEPT_ORG5='ALL' or :BLOCK1.P_Accept_Org5 = RA.accept_org5_code) and (:BLOCK1.P_ACCEPT_ORG6='ALL' or :BLOCK1.P_Accept_Org6 = RA.accept_org6_code) and (:BLOCK1.P_ACCEPT_ORG7='ALL' or :BLOCK1.P_Accept_Org7 = RA.accept_org7_code) and RA.approved_flag='Y' and RA.formal_flag='N' and RA.trans_id = (Select max(RA2.trans_id) from RA_CONTROL RA2 where RA.bureau_code=RA2.bureau_code and RA.document_o = RA2.document_no and RA.formal_flag='N' and RA2.approved_flag = 'Y') order by 4; </pre> <p>The Customer_No and Customer_Name are copies to the screen fields (P_Customer_No and Customer_Name), if the Agreement No</p>

Field Usage in RA502

Screen Name	Attribute	Specification
		(P_TWA_NO) is not ALL.
	Validation	Based on the LOV – the word ALL or a valid TWA Number. Error Message: “Invalid TWA Number.”
Customer No	Field Name	P_CUSTOMER_NO
	Attributes	Enterable, Required, Displayed
	Type	Varchar2(10)
	Default Value	ALL
	LOV	<pre> Select 'ALL' Customer_No, 'All Customers' Customer_Name , ' ' orderkey From dual UNION Select CC.customer_no, CC.customer_Name from Customer CC, RA_Control RA where RA.bureau_code = :BLOCK1.P_BUREAU_code and RAC.customer_no = CC.customer_No and (:BLOCK1.P_FUND_CODE = 'ALL' or :BLOCK1.P_FUND_CODE = RA.fund_code) and (:BLOCK1.P_ACCEPT_ORG1='ALL' or :BLOCK1.P_Accept_Org1 = RA.accept_org1_code) and (:BLOCK1.P_ACCEPT_ORG2='ALL' or :BLOCK1.P_Accept_Org2 = RA.accept_org2_code) and (:BLOCK1.P_ACCEPT_ORG3='ALL' or :BLOCK1.P_Accept_Org3 = RA.accept_org3_code) and (:BLOCK1.P_ACCEPT_ORG4='ALL' or :BLOCK1.P_Accept_Org4 = RA.accept_org4_code) and (:BLOCK1.P_ACCEPT_ORG5='ALL' or :BLOCK1.P_Accept_Org5 = RA.accept_org5_code) and (:BLOCK1.P_ACCEPT_ORG6='ALL' or :BLOCK1.P_Accept_Org6 = RA.accept_org6_code) and (:BLOCK1.P_ACCEPT_ORG7='ALL' or :BLOCK1.P_Accept_Org7 = RA.accept_org7_code) and (:Block1.P_STATUS = 'ALL' or :BLOCK1.P_STATUS = RA.RA_STATUS) and (:Block1.P_AGREEMENT_NO = 'ALL' or :BLOCK1.P_AGREEMENT_NO = RA.Agreement_no) and (:Block1.P_TWA_NO = 'ALL' or :BLOCK1.P_TWA_NO = RA.TEMP_WORK_AUTH_NO) and RA.trans_id = (Select max(RA2.trans_id) from RA_CONTROL RA2 where RA.bureau_code=RA2.bureau_code and RA.document_o = RA2.document_no and RA2.approved_flag = 'Y') order by 2; </pre>

Field Usage in RA502

Screen Name	Attribute	Specification
		The orderkey is not displayed in the LOV. It is only used for ordering the LOV
	Validation	Based on the LOV – the word ALL or a valid Customer No. Error Message: “Invalid Customer Number.”
Customer Name	Field Name	Customer_Name – Non-Database, Unlabeled
	Attributes	Not-Enterable, Navigable, Displayed
	Type	Varchar2(40)
	Value	Defaulted value “All Customers”. See LOV on Customer No (P_CUSTOMER_NO).
As of GL End Date	Field Name	P_GL_END_DATE_ASOF
	Attributes	Enterable, Required, Displayed
	Type	Date: Format “dd-Mon-yyyy”
	Default Value	None
	LOV	<p>Select AP.gl_end_date, AP.gl_status, YS.fiscal_year From aperiod AP, yearend_status YS where AP.gl_end_date <= sysdate and AP.gl_end_date between YS.beg_date and YS.end_date order by gl_end_date desc;</p> <p>The GL_End_Date and Fiscal_Year populate the P_GL_END_DATE_ASOF and FISCAL_YEARfield of the screen. The GL_Status is only displayed on the LOV.</p>
	Validation	<p>The validation is almost identical to the LOV except there is no restriction on AP.gl_end_date.</p> <p>Select AP.gl_end_date, YS.fiscal_year From aperiod AP, yearend_status YS where AP.gl_end_date <= sysdate and AP.gl_end_date between YS.beg_date and YS.end_date order by gl_end_date desc;</p> <p>Error Message: “Invalid GL End Date.”</p>
Fiscal Year	Field Name	FISCAL_YEAR
	Attributes	Not --Enterable, Navigable, Displayed, Drop Down List.
	Type	Number(4,0) Format 9999
	Default Value	None: Populated by the As of GL-End-Date (FISCAL_YEAR) validations
Order No Sequence	Field Name	P_ORDER_SEQ
	Attributes	Enterable, Required, Displayed
	Type	Varchar2(10). Pull down LOV with to values “ASCENDING” and “DESCENDING”

Field Usage in RA502

Screen Name	Attribute	Specification
	Default Value	ASCENDING
	LOV	Pull down LOV with to values “ASCENDING” and “DESCENDING

Upon entering the form all fields are defaulted except As-of-GL-End-Date (and Fiscal Year). As values are changed the in various fields the ability to change other fields must be modified. Therefore on the Post field trigger of each field enforce the following rules.

Interaction of Fields

Field	Triggering Action	Result
Bureau Code (P_BUREAU)	Real Change of values	Restore the default values (ALL) to Fund Code, Acceptance Orgs, Agreement Status, Agreement No, TWA No, and Customer No.
Fund Code (P_FUND_CODE)	Real Change of values	Restore the default values (ALL) to Agreement No, and TWA No.
Accept Org1 (P_ACCEPT_ORG1)	Value is ALL	Default P_ACCEPT_ORG2 through P_ACCEPT_ORG7 to “ALL” and set fields to be navigable but not enterable/updatable.
Accept Org1 (P_ACCEPT_ORG1)	Value is NOT ALL	Set P_ACCEPT_Org2 to be enterable/updatable.
Accept Org2 (P_ACCEPT_ORG2)	Value is ALL	Default P_ACCEPT_ORG3 through P_ACCEPT_ORG7)to “ALL” and set fields to be navigable but not enterable/updatable.
Accept Org2 (P_ACCEPT_ORG2)	Value is NOT ALL	Set P_ACCEPT_Org3 to be enterable/updatable.
Accept Org3 (P_ACCEPT_ORG3)	Value is ALL	Default P_ACCEPT_ORG4 through P_ACCEPT_ORG7)to “ALL” and set fields to be navigable but not enterable/updatable.
Accept Org3 (P_ACCEPT_ORG3)	Value is NOT ALL	Set P_ACCEPT_Org4 to be enterable/updatable..
Accept Org4 (P_ACCEPT_ORG4)	Value is ALL	Default P_ACCEPT_ORG5 through P_ACCEPT_ORG7)to “ALL” and set fields to be navigable but not enterable/updatable.
Accept Org4 (P_ACCEPT_ORG4)	Value is NOT ALL	Set P_ACCEPT_Org5 to be enterable/updatable.
Accept Org5 (P_ACCEPT_ORG5)	Value is ALL	Default P_ACCEPT_ORG6 through P_ACCEPT_ORG7)to “ALL” and set fields to be navigable but not enterable/updatable.
Accept Org5 (P_ACCEPT_ORG5)	Value is NOT ALL	Set P_ACCEPT_Org6 to be enterable/updatable.

Interaction of Fields

Field	Triggering Action	Result
Accept Org6 (P_ACCEPT_ORG6)	Value is ALL	Default P_ACCEPT_ORG7 to “ALL” and set fields to be navigable but not enterable/updatable.
Accept Org6 (P_ACCEPT_ORG6)	Value is NOT ALL	Set P_ACCEPT_Org7 to be enterable/updatable.
Agreement No (P_AGREEMENT_NO)	Value is ALL	Set P_TWA_NO (TWA No) to be enterable/updatable.
Agreement No (P_AGREEMENT_NO))	Value is NOT ALL	Set P_TWA_NO (TWA No) to be NOT enterable/updatable. Set P_TWA_NO = ALL Set P_CUSTOMER_NO (Customer No) to be Not enterable/updatable. Set P_CUSTOMER_NO = ALL and Customer Name = “All Customers”, .
TWA No (P_TWA_NO)	Value is ALL	Set P_AGREEMENT_NO (Agreement No) to be enterable/updatable.
TWA No (P_TWA_NO))	Value is NOT ALL	Set P_AGREEMENT_NO (Agreement No) to be NOT enterable/updatable. Set P_AGREEMENT_NO = ALL Set P_CUSTOMER_NO (Customer No) to be Not enterable/updatable. Set P_CUSTOMER_NO = ALL and Customer Name = “All Customers”,
Both P_AGREEMENT_NO and P_TWA_NO	Both Fields are ALL	Set P_CUSTOMER_NO (Customer No) to be enterable/updatable

2.1 Buttons

The following table specifies the action of the screen buttons.

Buttons

Screen Name	Specification
Close Window (X)	When the close window X is clicked, close the form (rollback) without validation of the current record.
Cancel	Same as Close Window (X).
Run Report	Validate that the As Of GL-End-Date is not null. (All of the other fields cannot be null). Execute the report using the parameters entered.

3. Report Format

This section provides the visual format of the report, describes the source of each of the elements on the report, and the sorting and subtotal breaks.

3.1 Report Field Attributes

The following table provides an overview of the data elements to be included on this report.

Table/Report Field Name	Description	Source
Cover Page		
Bureau Code	The Bureau Code specified on the Parameter Screen.	Value specified on the Parameter Screen.
Fund Code	The Fund Code specified on the Parameter Screen.	If the request is for all funds then the value printed will be “ALL” and the fund title is suppressed, otherwise the fund title of the specific fund is printed.
Acceptance Organization	The Acceptance Organization specified on the Parameter Screen.	If the request is for all acceptance organizations then the value “ALL” will be printed, otherwise the values entered will be printed. For example, if the selection criteria specifies org1=01 and org2=02 and all of the other values are “ALL” then the value displayed will be: “01-02-ALL-ALL-ALL-ALL-ALL”
Agreement Status	The Agreement Status specified on the Parameter Screen.	Value specified on the Parameter Screen.
Agreement Number	The Agreement Number specified on the Parameter Screen.	Value specified on the Parameter Screen.
TWA Number	The TWA Number specified on the Parameter Screen.	Value specified on the Parameter Screen.
Customer Number/Name	The Customer Number/Name specified on the Parameter Screen.	If a specific value is entered then the customer name will be displayed, otherwise the word “ALL” will be displayed.
As of GL End Date	The As of GL End Date specified on the Parameter Screen.	The As of GL End Date is the value the user entered in the selection criteria. This is NOT the same as the “AS OF” value printed on line 3 of the page header (“AS OF” date is the date that the data was copied from CFS into the data warehouse).
Order No. Sequence	The Order Number Sequence specified on the Parameter Screen.	Value specified on the Parameter Screen (either Ascending or Descending).

Table/Report Field Name	Description	Source
WIP Last Run Date	For each fund code, the WIP Last Run Date will display the Fund Code range of the RADG006 run (9999-9999) followed by the general ledger date of the run (GL-End-Date) followed by the date on which the RADG006 process was executed ("on DD-MON-YYYY").	
Report Detail		
Bureau	Bureau Code is a report specification parameter.	From the RADG002 tables (RA_CONTROL). The Bureau Name is from the BUREAU table.
Fund	Fund Code is an optional report specification parameter.	Fund Code comes from the RADG002 data. The fund title is from the FUND table, the current fiscal year is used to select the appropriate record.
Last WIP Run Date	The date on which the WIP was last run for this fund code.	Immediately after printing the Fund Code and Fund Title the Last WIP Run Date is printed. The ledger period of the RADG006 process and the date on which the process was run are displayed (WIP Run Date: GL Date: dd-mon-yyyy on dd-mon-yyyy). These values agree with the values of the Title Page.
Acceptance Organization	The acceptance organization codes are optional report specification parameters.	The Acceptance Organization codes come from the current RA_CONTROL (RADG002) record. Org Descriptions 1, 2, and 3 come from the ORG1, ORG2, and ORG3 tables respectively.
Agreement Status	The agreement status is an optional report specification parameter.	RA_STATUS in RA_CONTROL (RADG002)

Table/Report Field Name	Description	Source
Agreement/TWA No.	This field (actually 2 fields) are (optional) report specifications.	<p>From RA_CONTROL. If the agreement is formal, the Agreement No is printed; otherwise, the TWA No. is printed.</p> <p>The TWA No is on all records therefore one can select by TWA No even though the document is now a formal agreement.</p> <p>However, if the formal flag is Y, the value displayed will be the agreement number since the agreement is now formal.</p>
RADG002 Control #	The Control No that is assigned to the agreement by RADG002.	RA_CONTROL.DOCUMENT_NO
Highest Mod No.	The highest modification number.	<p>If the agreement is formal this is the MOD_NO value on the last RADG002 transaction (as of the AS OF GL End Date).</p> <p>Otherwise, it is the TWA_MOD_NO value on the last RADG002 transaction.</p>
Customer No./Name	Customer Number is an optional report specification.	<p>The Customer No. comes from the Customer No. field of RA_CONTROL.</p> <p>The Customer Name comes from the customer table.</p>
Period of Performance	The beginning and ending dates of the agreement.	From RA_CONTROL BEGIN_DATE and END_DATE.
Billing Type	The values are WIP (bill as you go) and ADV (advance).	From RA_CONTROL BILLING_TYPE.
Total RA Amount	The initial Reimbursable Agreement amount plus all approved modifications.	The CONTRACT_AMOUNT from the last approved RA_CONTROL (as of the As Of GL End Date).
Order No.	This is the text string that identifies the order in RADG003.	The ORDER_NO from the RA_ORDER_CONTROL table
RADG003 Control #	This is the unique identifier of RADG003 documents.	The DOCUMENT_NO field in RA_ORDER_CONTROL.
FCFY	The fund code fiscal year of the data.	<p>This field (FCFY or Fund_Code_Fiscal_Year) is in the transaction tables.</p> <p>RA_ORDER_CONTROL, RA_WIP_BAL_CONTROL, and the A/R tables.</p>
Project Code	Identifies the project being sponsored by the agreement. This is the key field used to associate costs to orders and agreements.	This field is in the transaction tables. RA_ORDER_CONTROL, RA_WIP_BAL_CONTROL, and the A/R tables.

Table/Report Field Name	Description	Source
Order Amount	The initial order amount plus all approved modifications.	<p>The CUMULATIVE_AMOUNT on RA_ORDER_DETAIL for the latest (ASOF GL-End-Date) approved order.</p> <p>The order amount is based on the last approved transaction recorded in the RADG003 screen prior or on the AS OF GL End Date of the report.</p>
Advance Collected	The amount of advances collected.	<p>This is the sum of the collection data (AR_RECV_COLLECTIONS, etc.) for an item type of "RMADV".</p> <p>The advance collected amount is the sum of all paid advances (collections) recorded through the A/R screens as of the AS OF GL End Date. These documents have an ITEM_TYPE value of "RMADV".</p>
Costs Billed to Date	Costs billed to the project as of the AS OF GL-End-Date	<p>This information comes from the WIP tables. In particular, the RA_WIP_BAL_CONTROL (RADG006) table.</p> <p>However, if the billing is canceled or adjusted, the amount of the adjustment is recorded in the RA_WIP_BAL_DETAIL_OTHER table where Source Type = 'CANCEL'.</p> <p>They represent all of the costs billed – manually or through the Auto-WIP process – as of the last running of the RADG006 process.</p>
Costs Collect. To Date	The "costs collected to date" is the total of all collections recorded for the agreement as of the AS OF GL End Date.	<p>The costs come from the A/R tables. These include both manual and autowip issued bills as well as miscellaneous receivables, if any.</p> <p>All collection amounts that have been recorded with a GL-End-Date less than or equal to the AS OF GL-End-Date.</p> <p>Note: Advances applied (RAADV) which are recorded in the system as negative amounts on the bill are reported as collections for the purpose of this calculation.</p>

Table/Report Field Name	Description	Source
Unreleased Costs	Costs that have been attributed to the agreement (order) but which have not been billed.	<p>The unreleased costs (to date) come from the WIP tables. In particular, they come from RA_WIP_BAL_CONTROL (RADG006) and represent all of the costs that have not been billed.</p> <p>It is possible for the field to be negative if the amount of costs billed manually exceeds the amount of costs incurred.</p>
Uncollected Advance	Advances that have been billed but have not been collected.	<p>The uncollected advance comes from the A/R tables.</p> <p>The difference between the total billed amount and the total collected amount for item type RMADV.</p> <p>Technically, miscellaneous collections (AR009) and never billed for this calculation we compute the billed amount as equal to the collected amount for miscellaneous collections.</p> <p>Therefore, non-zero amounts occur only when the amount is billed but not collected.</p>
Remaining Order Balance	The unused portion of the order.	This value is calculated as: Order Amount minus Costs Billed to date minus Unreleased Costs

3.2 Sort Order, Breaks, and Totals

Any agreement that does not have an order is included in the report. For these lines one only prints the Agreement information and a line with “** No Orders **” for the order information. Omit the Order summary.

That is, if the agreement does not have orders, then after printing the Agreement information print one line (without column titles) with the phrase “** No Orders **”. These lines are signaled by an ORDER_DOCUMENT_NO value of null and an ORDER_NO of “** No Orders **”. The next record will be a new agreement. However, after a “No Orders” agreement the Orders and Agreement Totals line is suppressed and only the line of equal signs (“====”) is printed.

If the agreement does have one or more orders then the current record has a non-null ORDER_DOCUMENT_NO value. After printing the agreement information begin a repeating frame of order document information listings. The current record contains the first order. Break change of Order Document No and print the Order Total. When a new agreement (or the end of the cursor) signals an agreement break print the agreement total and the line of equal signs (“====”).

The table below reflects the order in which the report will be sorted in ascending sequence unless other wise noted. The Break column reflects the type of break that will occur based on a change in the field name. For example, whenever the Bureau Code changes, a page break will occur. The Total column reflects if a total/subtotal will be provided whenever the value for that field changes.

Field Name in Sort Order Sequence	Break	Total
Bureau Code	Page	Yes
Fund Code	Page	Yes
Acceptance Organization	Page	Yes
Status	Page	No
Agreement/TWA No.	Division Line	Yes
RADG002 Control #		
Highest Mod No.		
Order No. (Ascending/Descending based on preference selected)	Total	Yes
RADG003 Control No.		
FCFY		
Project Code		

3.3 RA502 Report Format

CREATION RUN DATE: mm/dd/yyyy
REPORT_ID: RA502D
INSTANCE: DBINST.WORLD

COMMERCE BUSINESS SYSTEM
BUREAU NAME
REIMBURSABLE AGREEMENT SUMMARY REPORT

USER ID: OPS\$FIRLAS99
PAGE: 999999
AS OF: mm/dd/yyyy

BUREAU CODE: 99 - BUREAU DESCRIPTION 2XXXXXXXXX3XXXXXXXXXX4
FUND CODE: 9999 - FUND TITLE XXXXXXXX2XXXXXXXXX3XXXXXXXXXX4
ACCEPTANCE ORGANIZATION: XX-XX-XXXX-99-99-99-99
AGREEMENT STATUS: XXXXXX
AGREEMENT NUMBER: XXXXXXXX1XXXXXXXXXX2XXXXXXXXXX3
TWA NUMBER: XXXXXXXX1XXXXXXXXXX2XXXXXXXXXX3
CUSTOMER NUMBER: 99999999 - CUSTOMER NAME XXXXX2XXXXXXXXXX3XXXXXXXXXX4
AS OF GL END DATE: DD-MON-YYYY

ORDER NO. SEQUENCE: ASCENDING/DESCENDING

WIP LAST RUN DATE: 9999 - 9999: DD-MON-YYYY on DD-MON-YYYY
9999 - 9999: DD-MON-YYYY on DD-MON-YYYY
9999 - 9999: DD-MON-YYYY on DD-MON-YYYY

Version Data and/or Optional Report Message Data

NDW Refresh Status - SUCCESSFUL - START: mm/dd/yyyy hh:mi:ss - END: mm/dd/yyyy hh:mi:ss

CREATION RUN DATE: mm/dd/yyyy COMMERCE BUSINESS SYSTEM
REPORT_ID: RA502D
INSTANCE: DBINST.WORLD

USER ID: OPS\$FIRLAS99
BUREAU NAME
REIMBURSABLE AGREEMENT SUMMARY REPORT

PAGE: 999999
AS OF: mm/dd/yyyy

BUREAU: 99 - BUREAU DESCRIPTION 2XXXXXXXXX3XXXXXXXXX4
FUND: 9999 - FUND_TITLE XXXXXXXX2XXXXXXXXX3XXXXXXXXX4

LAST WIP Run Date: GL Date: dd-mon-yyyy on dd-mon-yyyy

ACCEPTANCE ORGANIZATION: XX-XX-XXXX-99-99-99-99 - ORG1 DESCRIPTION XX2XXXXXXXXX3XXXXXXXXX4 - ORG2 DESCRIPTION XX2XXXXXXXXX3XXXXXXXXX4 - ORG3 DESCRIPTION
AGREEMENT STATUS: XXXXXX

AGREEMENT/TWA NO.	RADG002 CONTROL#	HIGHEST MOD NO.	CUSTOMER NO/NAME	PERIOD OF PERFORMANCE	BILLING TYPE	TOTAL RA AMOUNT
XXXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX	99999999	9999	99999999-XXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX3XXXXXXXXXX4	DD-MON-YYYY - DD-MON-YYYY	XXXXXX	\$99,999,999.99

** NO Orders **

AGREEMENT/TWA NO.	RADG002 CONTROL#	HIGHEST MOD NO.	CUSTOMER NO/NAME	PERIOD OF PERFORMANCE	BILLING TYPE	TOTAL RA AMOUNT
XXXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX	99999999	9999	99999999-XXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX3XXXXXXXXXX4	DD-MON-YYYY - DD-MON-YYYY	XXXXXX	\$99,999,999.99

Source of information (not printed)

ORDER NO.	RADG003 CONTROL#	PROJECT FCFY CODE	ORDER AMOUNT	RADG003 EOM	A/R EOM (AS OF) ADVANCE COLLECTED	WIP Last WIP COSTS BILLED TO DATE	A/R EOM COSTS COLLECT.	WIP Last WIP UNRELEASED COSTS	A/R EOM (AS OF) UNCOLLECTED ADVANCE	Difference REMAINING ORDER BALANCE
XXXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX	99999999	9999	XXXXXXX	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
			XXXXXXX	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
			XXXXXXX	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
			XXXXXXX	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
TOTAL ORDER				\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
XXXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX	99999999	9999	XXXXXXX	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
			XXXXXXX	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
			XXXXXXX	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
ORDER TOTAL				\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
AGREEMENT TOTAL				\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99

AGREEMENT/TWA NO.	RADG002 CONTROL#	HIGHEST MOD NO.	CUSTOMER NO/NAME	PERIOD OF PERFORMANCE	BILLING TYPE	TOTAL RA AMOUNT
-------------------	---------------------	--------------------	------------------	-----------------------	-----------------	--------------------

XXXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX	99999999	9999	99999999-XXXXXXXX1XXXXXXXXXX2XXXXXXXXXX3XXXXXXXXXX4				DD-MON-YYYY - DD-MON-YYYY		XXXXXX	\$99,999,999.99	
ORDER NO.	RADG003 CONTROL#	PROJECT FCFY	CODE	ORDER	AMOUNT	ADVANCE COLLECTED	COSTS BILLED TO DATE	COSTS COLLECT. TO DATE	UNRELEASED COSTS	UNCOLLECTED ADVANCE	REMAINING ORDER BALANCE
XXXXXXXXXX1XXXXXXXXXX2XXXXXXXXXX	99999999	9999	XXXXXXX		\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99
			XXXXXXX		\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99	\$99,999,999.99

4. Selection Criteria and Report Execution

The RA502D report will select all approved Reimbursable Agreement documents meeting the selection criteria entered on the parameter screen. The high level criteria for selecting data for this report is as follows:

- Reimbursable Agreement documents will be selected with an APPROVED_DATE being less than or equal to the GL_END_DATE Parameter.
- Order, billing, collection, and cost data will be selected with a GL_END_DATE being less than or equal to the GL_END_DATE Parameter.
- The Reimbursable Agreement Status values are:
 - ACTIVE - ACTIVE TWA RECORD
 - CLOSED - TERMINATED AGREEMENT
 - OPEN - OPEN, ACTIVE AGREEMENT
 - VOID - INVALID AGREEMENT OR TWA

Normally, the selection criteria value of “ALL” would select agreements with any of the four values. However, for this report “ALL” selects only “ACTIVE” and “OPEN” documents. If the user enters the selection criteria of “CLOSED” or “VOID” they would receive a report on agreements with those status values.

There are some cases where an agreement has a null status value. The report shall always include these in the ALL or OPEN report but the status value of “null” will be printed.

There are two ways to generate the report.

1. The developer can use a single cursor that selects the header data (the Agreement data) and all of the detail data (the Orders, WIP Costs and A/R data) in one step. The developer can then format the reports as needed. Pictorially, the logic is as follows:

Agreement 1 ---- Date	Order Wip A/R 1
Agreement 1 ---- Date	Order Wip A/R 2
Agreement 2 ---- Date	Order WIP A/R 1
Agreement 3 ---- Date	Order WIP A/R 1
Agreement 3 ---- Date	Order WIP A/R 2
Agreement 4 ---- Date	No Order Wip A/R data
Agreement 5 ---- Date	Order WIP A/R 1

- The alternative is to use a temporary table and insert into the temporary table the Orders, WIP Costs and A/R data and format the data into records that can easily be selected as a sub cursor to the main cursor which is a select against the Agreement table. Pictorially, the logic is as follows:

Agreement 1 ---- Date	Order Wip A/R 1
	Order Wip A/R 2
Agreement 2 ---- Date	Order WIP A/R 1
Agreement 3 ---- Date	Order WIP A/R 1
	Order WIP A/R 2
Agreement 4 ---- Date	No Order Wip A/R data
Agreement 5 ---- Date	Order WIP A/R 1
	Order WIP A/R 2

The developer will use separate cursors to get the fund title, and the three organization titles.

4.1 Single Cursor Method

The following SQL query is the main query for the RA502D report. Note: the script was written in TOAD so the replacement parameters have a colon (":") prefix. These would be removed for Oracle Reports.

```
--** Main query for RA502D report
--** Some fields are not used -- they are present for analysis during development
--**

select * from
(
Select
  P.project_type                                --** Not Used
, RAC.gl_end_date_asof
, RAC.bureau_Code
, RAC.fund_code
--
, nvl(RAC.ra_status, 'Null')                    Ra_Status
, RAC.ra_status_date                            --** Not Used
--
, RAC.accept_org1_code
, RAC.accept_org2_code
, RAC.accept_org3_code
, RAC.accept_org4_code
, RAC.accept_org5_code
, RAC.accept_org6_code
, RAC.accept_org7_code
--
, RAC.trans_id                                  -- Not Used
, RAC.document_no
, RAC.formal_flag                              -- Not Used
```

```

,case when RAC.formal_flag='Y' then RAC.agreement_no
      else RAC.temp_work_auth_no end          agreement_twa_no
,case when RAC.formal_flag='Y' then RAC.Mod_no
      else RAC.twa_mod_no end                Mod_TWA_Mod_no
--
,RAC.customer_no
,CUST.customer_name
--
,RAC.begin_date
,RAC.end_date
--
,RAC.Contract_amount          -- Total RA Amount
,RAC.approved_flag            -- Not Used
,RAC.active_flag              -- Not Used
,RAC.billing_type
--
,ROC.ORDER_DOCUMENT_NO        -- Radg002 #
,ROC.ORDER_NO

,ROC.FUND_CODE_FISCAL_YEAR
,ROC.project_code
,ROC.order_amount              order_amount
,ROC.active_flag              order_active_flag
,sum(NVL(AMT.Costs_Billed_todate,0))      Costs_Billed_Todate
,sum(NVL(AMT.Unreleased_Costs,0))          Unreleased_Costs
,sum(NVL(AMT.Advance_Collected,0))        Advance_Collected
,sum(NVL(AMT.costs_Collected_todate,0))   Costs_Collected_todate
,sum(NVL(AMT.uncollected_advance,0))      Uncollected_Advance
from project P, customer CUST, ndw_ra_control_asof_v RAC
,( select ROC2.gl_end_date_asof
      ,ROC2.bureau_code
      ,ROC2.fund_code
      ,ROC2.ra_document_no
      ,ROC2.ORDER_DOCUMENT_NO        -- Radg002 #
      ,ROC2.ORDER_NO

      ,ROC2.FUND_CODE_FISCAL_YEAR
      ,ROC2.project_code
      ,ROC2.active_flag
      ,sum(nvl(ROC2.cumulative_amount,0)) order_amount
from ndw_ra_order_c_detail_asof_v ROC2
group by
      ROC2.gl_end_date_asof
      ,ROC2.bureau_code
      ,ROC2.fund_code
      ,ROC2.ra_document_no
      ,ROC2.ORDER_DOCUMENT_NO        -- Radg002 #
      ,ROC2.ORDER_NO

      ,ROC2.FUND_CODE_FISCAL_YEAR
      ,ROC2.project_code
      ,ROC2.active_flag
) ROC
--
, (select
  BUREAU_CODE
, FUND_CODE
, PROJECT_CODE
, FUND_CODE_FISCAL_YEAR
, GL_END_DATE_ASOF
, GL_END_DATE
, ORDER_DOCUMENT_NO
--
, 0
, 0
--
, decode(item_type, 'RMADV',1,0)*
  PRINC_COLLECT          Advance_Collected  --2
, decode(item_type, 'RMWIP',1,'RAWIP',1,0)*
  PRINC_COLLECT          costs_Collected_todate--2
, decode(item_type, 'RMADV',1,0)*
  ( PRINC_BILLED_NET - PRINC_COLLECT )      uncollected_advance  --2
from ndw_ra502_503_rcv_asof_v
UNION ALL
select

```

```

        BUREAU_CODE
    ,    FUND_CODE
    ,    PROJECT_CODE
    ,    FUND_CODE_FISCAL_YEAR
    ,    GL_END_DATE_ASOF
    ,    GL_END_DATE
    ,    ORDER_DOCUMENT_NO
--
    ,    BILLED_COST
    ,    UNRELEASED_COST
--
    ,    0
    ,    0
    ,    0
from ndw_ra502_503_wip_asof_v) Amt
where RAC.gl_end_date_asof = :P_gl_end_date_asof
    and AMT.gl_end_date_asof = :P_gl_end_date_asof
--
    and CUST.customer_no = RAC.customer_no
    and ROC.gl_end_date_asof = RAC.gl_end_date_asof
    and ROC.ra_document_no = RAC.DOCUMENT_NO
--
    and P.bureau_code = ROC.bureau_code
    and P.project_code = ROC.project_code
--
    and AMT.order_document_no(+) = ROC.order_document_no
    and AMT.bureau_code(+) = ROC.bureau_code
    and AMT.fund_code(+) = ROC.fund_code
    and AMT.fund_code_fiscal_year(+) = ROC.fund_code_fiscal_year
    and AMT.project_code(+) = ROC.project_code
    and (RAC.bureau_code, RAC.fund_code) in
        (select bureau_code,fund_code
         from fund FU
         where FU.bureau_code = :P_bureau_code
           and FU.fund_code =
decode(:p_fund_code,'ALL',FU.fund_code,:p_fund_code)
           and FU.fund_type like 'REIMB%'
           and FU.fund_type <> 'REIMBB'
        )
--
    and RAC.accept_org1_code =
        decode(:P_accept_org1,'ALL', RAC.accept_org1_code,:p_accept_org1)
    and RAC.accept_org2_code =
        decode(:P_accept_org2,'ALL', RAC.accept_org2_code,:p_accept_org2)
    and RAC.accept_org3_code =
        decode(:P_accept_org3,'ALL', RAC.accept_org3_code,:p_accept_org3)
    and RAC.accept_org4_code =
        decode(:P_accept_org4,'ALL', RAC.accept_org4_code,:p_accept_org4)
    and RAC.accept_org5_code =
        decode(:P_accept_org5,'ALL', RAC.accept_org5_code,:p_accept_org5)
    and RAC.accept_org6_code =
        decode(:P_accept_org6,'ALL', RAC.accept_org6_code,:p_accept_org6)
    and RAC.accept_org7_code =
        decode(:P_accept_org7,'ALL', RAC.accept_org7_code,:p_accept_org7)
    and ( nvl(RAC.RA_Status,'OPEN') = :P_Status
        or ( :P_Status = 'ALL' and nvl(RAC.RA_Status,'OPEN') <> 'VOID' )
        )
    and RAC.document_no =
        decode(:P_Agreement_No,'ALL',RAC.document_no, :P_Agreement_NO)
    and (:P_TWA_No = 'ALL'
        or :P_TWA_No = RAC.Temp_Work_Auth_no )
    and RAC.customer_no =
        decode(:P_Customer_No,'ALL',RAC.customer_no, :P_customer_no)
group by
    P.project_type
    ,RAC.gl_end_date_asof
    ,RAC.bureau_Code
    ,RAC.fund_code
--
    ,RAC.ra_status
    ,RAC.ra_status_date
--
    ,RAC.accept_org1_code
    ,RAC.accept_org2_code
    ,RAC.accept_org3_code

```

```

,RAC.accept_org4_code
,RAC.accept_org5_code
,RAC.accept_org6_code
,RAC.accept_org7_code
--
,RAC.trans_id
,RAC.document_no
,RAC.formal_flag
,case when RAC.formal_flag='Y' then RAC.agreement_no
      else RAC.temp_work_auth_no end
,case when RAC.formal_flag='Y' then RAC.Mod_no
      else RAC.twa_mod_no end
--
,RAC.customer_no
,CUST.customer_name
--
,RAC.begin_date
,RAC.end_date
--
,RAC.Contract_amount
,RAC.approved_flag
,RAC.active_flag
,RAC.billing_type
--
,ROC.ORDER_DOCUMENT_NO
,ROC.ORDER_NO
,ROC.FUND_CODE_FISCAL_YEAR
,ROC.project_code
,ROC.order_amount
,ROC.active_flag
UNION ALL
Select
  'Unknown'                                Project_Type--** Not Used
,RAC.gl_end_date_asof
,RAC.bureau_Code
,RAC.fund_code
--
,nvl(RAC.ra_status,'Null')                 Ra_Status
,RAC.ra_status_date                       --** Not Used
--
,RAC.accept_org1_code
,RAC.accept_org2_code
,RAC.accept_org3_code
,RAC.accept_org4_code
,RAC.accept_org5_code
,RAC.accept_org6_code
,RAC.accept_org7_code
--
,RAC.trans_id                             -- Not Used
,RAC.document_no
,RAC.formal_flag                           -- Not Used
,case when RAC.formal_flag='Y' then RAC.agreement_no
      else RAC.temp_work_auth_no end       agreement_twa_no
,case when RAC.formal_flag='Y' then RAC.Mod_no
      else RAC.twa_mod_no end               Mod_TWA_Mod_no
--
,RAC.customer_no
,CUST.customer_name
--
,RAC.begin_date
,RAC.end_date
--
,RAC.Contract_amount                       -- Total RA Amount
,RAC.approved_flag                         -- Not Used
,RAC.active_flag                           -- Not Used
,RAC.billing_type
--
, to_number(null)                          ORDER_DOCUMENT_NO
, '** No Orders **'                        ORDER_NO
, 0                                         FUND_CODE_FISCAL_YEAR
, null                                     project_code
, 0                                         order_amount
, null                                     order_active_flag
, 0                                         Costs_Billed_Todate
, 0                                         Unreleased_Costs

```

```

, 0
, 0
, 0
from customer CUST, ndw_ra_control_asof_v RAC
where RAC.gl_end_date_asof = :P_gl_end_date_asof
--
and CUST.customer_no = RAC.customer_no
--
and (RAC.bureau_code, RAC.fund_code) in
(select bureau_code, fund_code
from fund FU
where FU.bureau_code = :P_bureau_code
and FU.fund_code =
decode(:p_fund_code, 'ALL', FU.fund_code, :p_fund_code)
and FU.fund_type like 'REIMB%'
and FU.fund_type <> 'REIMBB'
)
--
and
RAC.accept_org1_code =
decode(:P_accept_org1, 'ALL', RAC.accept_org1_code, :p_accept_org1)
and RAC.accept_org2_code =
decode(:P_accept_org2, 'ALL', RAC.accept_org2_code, :p_accept_org2)
and RAC.accept_org3_code =
decode(:P_accept_org3, 'ALL', RAC.accept_org3_code, :p_accept_org3)
and RAC.accept_org4_code =
decode(:P_accept_org4, 'ALL', RAC.accept_org4_code, :p_accept_org4)
and RAC.accept_org5_code =
decode(:P_accept_org5, 'ALL', RAC.accept_org5_code, :p_accept_org5)
and RAC.accept_org6_code =
decode(:P_accept_org6, 'ALL', RAC.accept_org6_code, :p_accept_org6)
and RAC.accept_org7_code =
decode(:P_accept_org7, 'ALL', RAC.accept_org7_code, :p_accept_org7)
and ( nvl(RAC.RA_Status, 'OPEN') = :P_Status
or ( :P_Status = 'ALL' and nvl(RAC.RA_Status, 'OPEN') <> 'VOID' )
)
and RAC.document_no =
decode(:P_Agreement_No, 'ALL', RAC.document_no, :P_Agreement_NO)
and (:P_TWA_No = 'ALL'
or :P_TWA_No = RAC.Temp_Work_Auth_no )
and RAC.customer_no =
decode(:P_Customer_No, 'ALL', RAC.customer_no, :P_customer_no)
and not exists (select 'x' from ndw_ra_order_control_asof_v ROC
where ROC.ra_document_no = RAC.document_no
and ROC.approved_flag = 'Y'
and ROC.gl_end_date_asof =
:p_gl_end_date_asof
)
)
order by bureau_code, fund_code
, accept_org1_code, accept_org2_code, accept_org3_code
, accept_org4_code, accept_org5_code, accept_org6_code
, accept_org7_code
, RA_Status
, agreement_twa_no
, document_no
, MOD_TWA_MOD_NO
, decode(:P_Order_Seq, 'ASCENDING', order_no, '')
, decode(:P_Order_Seq, 'ASCENDING', '', order_no) desc
, order_document_no
, fund_code_fiscal_year
, project_code;

```

The query uses the parameters P_Bureau, P_Fund_Code, P_Accept_org1 through P_Accept_Org7, P_gl_end_date_Asof, P_Status, P_Agreement_No, P_TWA_No, P_Customer_No, and P_Order_Seq to select the date (See the **Summary of the Field Usage in RA502** table).

The ASOF views for Reimbursable Agreements (RADG002: RA_Control_Asof_V) and a summary of the Reimbursable Orders (RADG003: RA_Order_C_Detail_Asof_V) are joined.

The summary of Reimbursable Order (RADG003: RA_Order_C_Detail_Asof_V) is joined to the receivable data and the WIP data with an outer join to insure that the order is reported even if there is no receivable information, etc.

4.2 Temporary Table Method (alternative method)

The RA502D report can be generated from a single select statement as described earlier. However, one can use a temporary table (an Oracle Temporary table) and a stored procedure to perform the same process. The advantage of the temporary table method is that it follows the same logic as the RA503D report and it is less is easier to tune – if necessary.

An Oracle Temporary table is generated by the script in **RA502D_REPORT_STAGING.tbl**. There is an index on Program_Level and RA_Document_No.

The Program_Level field is used to identify inserted data at each step. Program levels 1 through 3 are reserved for raw insert data. Program level 11 is reserved for the report generation.

The following table identified the step and the companion SQL file.

Step	Program Level	File	Action
1			Delete the data in RA502D_REPORT_STAGING
2	1	Ra502d1.sql	Insert the A/R data into RA502D_REPORT_STAGING
3	2	Ra502d2.sql	Insert the WIP data into RA502D_REPORT_STAGING
4		Ra502d2a.sql	Update the Program Level 1 and 2 data, inserting the missing Order Document information.
5	3	Ra502d3.sql	Insert the Reimbursable Order Information (the RADG003 information).
6	11	Ra502d11.sql	Using Program Levels 1, 2, and 3 summarize the data and form the Program_Level 11 data sets which has all of the column of the Order reporting section of the report. Compute the Remaining Order Balance as: Order Amount - (Cost Billed todate + Unreleased Cost)
7		Ra502d10.sql	This cursor selects All of the agreement data satisfying the criteria specified. For each agreement use sub-cursors to get the Fund Title and Org Title information (see above).
7a		Ra502d10a.sql	Execute this cursor to product the Order Wip A/R information using the RAC.document_no (RA_DOCUMENT_NO). If there is no information print the "*** No Orders ***" line and the line of

			equals signs ("=====") and continue with the next agreement.
--	--	--	--

4.3 Getting Org Description, Fund Title

The cursors for getting Org Description are: (“Current_” represents the current values of the main cursor.):

```

select org1_descr from org1
where bureau_code = :current_Bureau_code
and org1_code = :current_org1_code

select org2_descr from org2
where bureau_code = :current_Bureau_code
and org1_code = :current_org1_code
and org2_code = :current_org2_code

select org3_descr from org3
where bureau_code = :current_Bureau_code
and org1_code = :current_org1_code
and org2_code = :current_org2_code
and org3_code = :current_org3_code

```

The cursors for getting Fund Title is the following. Note: In theory we should have FCFY to select the Fund Title but we can use the latest Fund Record. Currently, NOAA has only No-Year funds for reimbursable agreements so there will be only one title. However, if annual funds are used, we will select title from the latest valid record.

```

select FU.Fund_Title from Fund FU
where FU.bureau_code = :P_Bureau_code
and FU.Fund_code = :current_fund_code
and FU.begin_date =
(select max(begin_date)
from Fund F2
where F2.bureau_code = FU.bureau_code
and F2.fund_code = FU.fund_code
and FU.begin_date <= :p_gl_end_date_asof
):

```

To compute the “WIP LAST RUN DATE” information which is printed on the page header after the Fund Title, use the following query.

```

Select  to_char(RBL.fund_code_from,'0000') || '-' ||
        to_char(RBL.fund_code_to,'0000') || ':' ||
        to_char(RBL.gl_end_date,'DD-Mon-YYYY') || ' on ' ||
        to_char(RBL.finish_time,'DD-Mon-YYYY')
from ra_batch_log_asof_v RBL
where RBL.gl_end_date_asof = :p_gl_end_date_asof
and RBL.bureau_code = curr_bureau_code
and Curr_fund_code between
        RBL.fund_code_from and RBL.fund_code_to
and FU.fund_type like 'REIMB%'
and FU.fund_type <> 'REIMBB'

```

4.4 Title Page

To compute the “WIP LAST RUN DATE” information use the following query:

```

Select  distinct to_char(RBL.fund_code_from,'0000') || '-' ||
              to_char(RBL.fund_code_to  ,'0000') || ':' ||
              to_char(RBL.gl_end_date, 'DD-Mon-YYYY') || ' on ' ||
              to_char(RBL.finish_time, 'DD-Mon-YYYY')
from ra_batch_log_asof_v RBL,  FUnd FU
where RBL.gl_end_date_asof = :p_gl_end_date_asof
      and RBL.bureau_code = :p_bureau_code
      and RBL.bureau_code = FU.bureau_code
      and FU.fund_code between RBL.fund_code_from and RBL.fund_code_to
      and FU.fund_type like 'REIMB%'
      and FU.fund_type <> 'REIMBB'
      and FU.fund_code =
decode(:p_fund_code, 'ALL', FU.fund_code, :p_fund_code)

```

Note: The distinct clause is required since it is possible the 2 fund codes are on the same RA_BATCH_LOG record.

5. Data Warehouse Structure Support

The following CBS source tables will need to be snapshot into the data warehouse to support the RA502D report:

- RA_BATCH_LOG
- RA_CONTROL
- RA_DETAIL
- RA_ORDER_CONTROL (will be snapshot as part of RA504D promotion).
- RA_ORDER_DETAIL
- RA_WIP_BAL_CONTROL
- RA_WIP_BAL_DETAIL_COST (although not required for this report, this table will be snapshot for reference and potentially other reports.)
- RA_WIP_BAL_DETAIL_OTHER
- RA_WIP_PROCESS (will be snapshot as part of RA504D promotion).

6. Views

Several of the Reimbursable Agreement reports require the selection of data as of a specific date – the end of a ledger period. For selecting data from the ledger one can simply select all transactions with the GL-End-Date <= As-Of-Date.

However, for many tables the last record in the table that has a GL-End-Date <= As-Of-Date is required. Therefore, a series of views have been created that contain the field GL-End-Date-ASOF. This field is created by joining the data tables to the APERIOD table. The GL-End-Date of the APERIOD table becomes the GL-End-Date-ASOF column, and the GL-End-Dates of the other tables remain defined as GL-End-Date fields. This simplifies a number of queries.

For example, the Reimbursable Agreement document view selects data from the RA_CONTROL record which is the base table for the RADG002 screen. There is no GL-End-Date field in the RA_CONTROL table, but as noted in the report requirements the Approved-Date column is to be used as a substitute for the GL-End-Date field.

The following script defines the view RA_CONTROL_ASOF_V for the Reimbursable Agreement documents:

```
Create or replace view NDW_RA_CONTROL_ASOF_V
as
--
select AP.gl_end_date   gl_end_date_asof, RAC.*
  from ra_control RAC, Aperiod AP
 where RAC.approved_flag = 'Y'
       and RAC.approved_date <= AP.gl_end_date
       and RAC.trans_id = (select max(RAC2.trans_id)
                           from ra_control RAC2
                          where RAC2.document_no = RAC.document_no
                             and RAC2.approved_date <= AP.gl_end_date
                             and RAC2.approved_flag = 'Y');
```

At any point in time to find the current active (last approved) agreement record, the sub-query must be used to find the last approved records as of the GL-End-Date that limits the selecting of data. All of the fields of the RA_CONTROL records are selected and one field from the APERIOD record, the GL-End-Date alias GL-End-Date-ASOF.

Suppose that an agreement (Document No 600) is recorded in January (20-Jan-2006) for \$200 and is changed in March (06-Mar-2006) to \$500 (Contract-Amount). The following query will yield the following results:

```
Select Document_No, Contract_Amount, Approved_date
  from NDW_ra_control_asof_v
 where Document_No = 500
       and GL_End_Date_ASOF = :parameter_value
```

Parameter GL-End-Date- ASOF	Document No	Contract Amount	Approved Date
31-Dec-2005	No data found		
31-Jan-2006	600	\$200	20-Jan-2006
28-Feb-2006	600	\$200	20-Jan-2006
31-Mar-2006	600	\$500	06-Mar-2006
30-Apr-2006	600	\$500	06-Mar-2006

This view and all of the views that contain this logic to create a GL-End-Date-ASOF column must specify a restriction of GL-End-Date-ASOF using the equality condition. If the restriction on GL-End-Date-ASOF is omitted in the previous statement then the query will return one row for each month in the APERIOD table starting with 31-Jan-2006.

View Name	Purpose
NDW_RA_CONTROL_ASOF_V	Selects the appropriate RA_CONTROL (RADG002) record for the given As of Date.
NDW_RA_ORDER_CONTROL_ASOF_V	Select the appropriate RA_ORDER_CONTROL (RADG003) records for the given As of Date.
NDW_RA_ORDER_C_DETAIL_ASOF_V	Select the appropriate RA_ORDER_CONTROL and RA_ORDER_DETAIL records for the given As of Date. Uses NDW_RA_Order_Control_Asof_V
NDW_RA_BATCH_LOG_ASOF_V	This joins the RA_BATCH_LOG table to the associated RA_WIP_PROCESS record. One table contains the Bureau, Fund and GL-End-Date information (the RADG004 run) and the other contains the Batch ID of the last recording of the agreement data – the RADG006 run.
NDW_RA_RECV_BI_ADJ_COL_ASOF_V 0	Joins and unions data in the accounts receivable tables so that the information for the amount billed, adjusted and collected can be drawn from a single database object (view). The receivable number and item type are key values for this view. Note: this view treats advances applied items (item type RAADV) and payments of invoices (collections). See the view for details. This view is used by the RA_Recv_Bill_Adj_Coll_Asof_V view.

View Name	Purpose						
NDW_RA_RECV_BI_ADJ_COL_ASOF_V	<p>Summaries the data in NDW_RA_Recv_Bi_Adj_Col_Asof_V0 by Recv_id and Item_Type. Miscellaneous collections are treated as the transaction was billed and paid in full so that Billed receivables and Miscellaneous collections can be intermixed and treated with the same logic.</p> <p>The item type are</p> <table> <tr> <td>RMADV</td><td>Advances</td></tr> <tr> <td>RMWIP</td><td>Manual billings</td></tr> <tr> <td>RAWIP</td><td>Autowip (RADG006) transactions.</td></tr> </table>	RMADV	Advances	RMWIP	Manual billings	RAWIP	Autowip (RADG006) transactions.
RMADV	Advances						
RMWIP	Manual billings						
RAWIP	Autowip (RADG006) transactions.						
NDW_RA502_503_REVC_ASOF_V	<p>Joins the data in RA_Recv_Bill_Adj_Coll_Asof_V to the RA_ORDER_CONTROL table to summarize the data by reimbursable ORDER_DOCUMENT_NO.</p> <p>This data is in the form that is useful to the RA502 and RA503 reports.</p>						
NDW_RA502_503_WIP_ASOF_V	<p>Using RA_WIP_BAL_CONTROL and RA_WIP_BAL_DETAIL_OTHER (RADG006 generated data), computes the Billed Cost To-Date and Unreleased Cost To-Date amounts.</p> <p>This data is in the form that is useful to the RA502 and RA503 reports</p>						

7. Navigator Menu Parameters

The Data Warehouse Navigator menu has a feature to store parameters for a report so that values that might change do not need to be hard coded in a report. The RA502D report will select data only for Fund Types that are like “REIMB%” but not equal to “REIMBB”. This results in only 3 fund types that meet this criterion:

- REIMB
- REIMB6
- REIMBE

An entry will be made in the CAMS_MENU_PARAMETER table to specify the Fund Types to be included in this report:

- PARAMETER_NAME = REIMB_FUND_TYPES
- PARAMETER_VALUE = REIMB,REIMB6,REIMBE

Once created, any change to the Fund Types that should be selected can be updated by the Navigator Menu Administrator and will not require a code promotion.

8. Navigator Report Header and Cover Page Definitions

The data warehouse follows a standard convention for report headers and cover pages. The table below describes each of the fields and how they are populated.

Table/Report Field Name	Description	Source
Header Data		
Creation Run Date	The date on which the report was run.	System Date
Commerce Business System	Name of system from which reports are generated.	VALUE from NDW_DEFAULTS table where ITEM_NAME = 'SYSTEM_NAME'.
User ID	Oracle User ID of the user who requested the report.	Oracle User Account
Report ID	A unique identifier for the report.	Report IDs are hard coded in the report code.
Bureau Name	The name of the bureau.	BUREAU BUREAU_NAME where B.BUREAU_CODE = bureau code selected on parameter screen.
Page	The number of the report page.	Sequential number generated through the report.
Instance	The SID of the database instance against which the report is run.	Determined from the database instance.
Report Title	The title of the report.	Report Titles are hard coded in the report code.
As Of	The date as of when the data was refreshed (last snapshot) in the data warehouse.	SNAPSHOT_AS_OF_DATE from the NDW_REFRESH_STATUS table where the REFRESH_STATUS_TYPE = 'CURRENT'.
Cover Page		
<i>Parameter Data</i>	This is a series of fields that is tailored for each report to reflect the parameters available and the values selected for producing this report.	Report dependant.
<i>Report Notes and Version Data</i>	The data warehouse provides the capability for notes to be stored and presented on the cover sheet. Multiple note lines (2000 bytes) can be specified for each report and will be presented in NOTE_LINE_NO sequence.	NOTE from NDW_REPORT_NOTES table where REPORT_ID = report identifier of the report being generated.

Table/Report Field Name	Description	Source
NDW Refresh Status Line	<p>A standard line that reflects the status of the data in the data warehouse at the time the report was generated. The format of this line is:</p> <p>NDW REFRESH STATUS – <i>status</i> – START: <i>MM/DD/YYYY</i> <i>HH:MI:SS</i> – END: <i>MM/DD/YYYY</i> <i>HH:MI:SS</i></p> <p>If the status reflects anything other than SUCCESSFUL, the data within the report may be suspect and the report should be regenerated.</p>	NDW_REFRESH_TERMINATION_STATUS, NDW_REFRESH_START_TIME, and NDW_REFRESH_END_TIME from NDW_REFRESH_STATUS where the REFRESH_STATUS_TYPE = ‘CURRENT’.